AMENDMENTS TO THE SPECIFICATION

Please replace paragraph [001] with the following amended paragraph:

[001] The invention generally relates to the field of data security. More specifically the invention relates to security administration for groups of data items.

Please replace paragraph [002] with the following amended paragraph:

[002] When handling information it is often desirable to limit access to specific portions of the information such that the specific portions are only accessible to certain authorized users. When information is contained in physical documents (e.g., printed [[book]] books or ledgers), those documents can be secured using physical access controls such as locks and document custodians. However, in today's world, large amounts of information are stored in the form of digital data. Digital data is easily created, modified, copied, transported and deleted, which has resulted in the proliferation of vast amounts of digital data existing in a myriad of locations. Similar to physical documents, it is often desirable to limit access to portions of digital data. However, the sheer amount of digital data and ease of creating, copying, transporting, modifying, and deleting digital data make securing digital data challenging.

Please replace paragraph [007] with the following amended paragraph:

[007] In contrast, the security in [[data base]] database systems is done by assigning ACLs to entire columns. Assigning ACLs to entire columns allows searches to be performed more efficiently because a single ACL can be accessed to determine security permissions for an entire column of data. A database can be configured such that searches can only be performed on a column when a user has the appropriate security permissions to access the column. Thus, there

is no need to check permissions for each element within the table. One drawback of column based assignment of security permissions is that the granularity may be too coarse for some applications. For example while most of the digital data in a column representing a digital address book entry may be suitable for general access, it may be desirable to restrict access to some of the digital data such as Social Security numbers or other types of sensitive information. However, when an ACL is assigned to an entire column, security permissions cannot vary between different items in the column. Thus, there may be no way to limit access to a Social Security number without also similarly limiting access to address and telephone number.

Please replace paragraph [011] with the following amended paragraph:

[011] In another embodiment, of the invention a computer system delegates administrative rights to principals. The computer system comprises a volume that stores a number of items is divided into at least one non-overlapping zone. Each item resides in a zone from among the at least [[one-non overlapping]] one non-overlapping zone. By each item being in a zone, administrative rights can be delegated at an appropriate granularity that is finer than an entire database table but yet coarse enough so as to not require delegation for each item. The zones each have one or more principals with administrative rights. The computer system identifies first items in the main zone.

Please replace paragraph [021] with the following amended paragraph:

[021] The present invention extends to methods, systems, and computer program [[product]] products for zone based security administration for data items. In one embodiment, a computer system determines security rights to at least a portion of a data item included in a security zone.

That portion of a data item is specified through an element path such that security rules need not be applied at a cell level. In another embodiment of the invention, computer system delegates administrative rights, (i.e. the ability to change the security [[of]] to at least a portion of a data item) to principals. Each item resides in a zone from among the at least [[one-non overlapping]] one non-overlapping zone. By each item being in a zone, administrative rights can be delegated at an appropriate granularity that is finer than an entire database table but yet coarse enough so as to not require delegation for each item.

Please replace paragraph [025] with the following amended paragraph:

[025] A token 128 specifies information that allows the authentication module 146 to determine the identity of the principal presenting the token. For example in the present example, token 128 contains information that identifies a principal, namely application 122. The process of verifying the identity of a principal is often referred to as authenticating. The authentication module [[148]] 146 may have access to a database of authentication information to compare with the token. In one embodiment of the invention, this authentication database is stored in the storage 102. Once the principal has been authenticated, rights can be determined for the principal by examining the rights specified for that principal by consulting an access control list (ACL).

Please replace paragraph [030] with the following amended paragraph:

[030] In one embodiment of the invention an ACL 130 can be created by an administrative principal that has administrative rights over the items in data store 104 and in method store 112. The administrative principal may be one of a plurality of principals that have administrative

rights. In one embodiment of the invention, the plurality of principals that have administrative rights is the same for all items existing in a security zone as explained in more detail below.

Please replace paragraph [045] with the following amended paragraph:

[045] Method 500 includes an act accessing authentication information that indicates the identity of a principal has been verified (act 502). Act 502 can include a computer system accessing authentication information that indicates the identity of a principal has been verified. For example, a computer system in network architecture 100 can access authentication information provided by authentication module 146 or authentication information stored in cache 120. In one embodiment of the invention, act 502 is performed by an authentication module [[144]] 146 such as the authentication module shown in Figure 1.

Please replace paragraph [046] with the following amended paragraph:

[046] The authentication module may receive a token such as token 128 or token 140 from a principal. The authentication module then compares the information in the token to information that the authentication module [[144]] 146 has available to it to verify the identity of a principal. This information may be available in a database that contains authentication information. Several different types of tokens exists such as passwords, encrypted strings, physical keys such as smart cards, biometric keys such as fingerprints, voice analysis, etc. Two particular tokens that may be used are windows tokens and authenticated XrML license sets. Act 502 may be further performed by consulting a cache entry such as an entry in cache 120 shown in Figure 1. Specifically, in one embodiment of the invention, once a token has been authenticated by the authentication module [[144]] 146, information can be placed into the cache

120 such that subsequent authenticating of a principal in the same session can now be done by consulting the entry in cache.

Please replace paragraph [048] with the following amended paragraph:

[048] For example, if a security rule granted rights for item 312 in Figure 3, the element argument would specify the path to item 312 such as 306.308.312. This path includes a topmost item, item 306, an item that depends from item 306, item 308, and finally the item for which the security applies, item 312 which depends from item 308. In another embodiment of the invention a security rule may specify an element that is a complex element that comprises a plurality of attributes. In this case access can be granted to all attributes of the element. For example if the element path is 306.310, [[than]] then a principal would have rights to element 310 as well as all elements that depend from element 310 including elements 316 through 322.

Please replace paragraph [052] with the following amended paragraph:

[052] For example, a rule may specify a set of principals including all network administrators while excluding a particular administrator. If the particular network administrator excluded by the deny ACE is granted rights in a different security rule that are the same [[as in]] as those in a security rule from which the networks administrator was excluded, then the network administrator will have the specified security rights. For example, if one security rule specifies that all administrators except X have rights to item Y, and a second security rule specifies that administrator X has rights to item Y, then administrator X has rights to item Y. To remove administrator X's rights to item Y a deny ACE can be used to exclude administrator X in every rule granting rights to item Y. The security rule in one embodiment of the invention is a grant,

meaning that rights can be granted. Rights cannot be taken away from the principal by creating a deny rule, but rather all of the grant rules must be modified or removed to disallow a principal from certain rights.

Please replace paragraph [062] with the following amended paragraph:

[062] With reference to Figure 6, an exemplary system for implementing the invention includes a general purpose computing device in the form of a conventional computer 620, including a processing unit 621, a system memory 622, and a system bus 623 that couples various system components including the system memory 622 to the processing unit 621. The system bus 623 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory includes read only memory (ROM) 624 and random access memory (RAM) 625. A basic input/output system (BIOS) 626, containing the basic routines that help transfer information between elements within the computer [[20]] 620, such as during start-up, may be stored in ROM 624.